



“Proyecto Academia Musical JACQUIN”

Diseño de patrones de software

Aprendiz:

Manuel Prudencio Pertuz Pérez

Servicio Nacional de Enseñanza – SENA

Centro Minero Regional Boyacá.

2977343 – Análisis Y Desarrollo de Software.

Instructor:

Luis Edilberto Díaz Sandoval

Julio - 2025.



Tabla de contenido

Introducción	3
Incorporación de patrones de diseño	4
Vista de componentes	4
Vista de despliegue	5
Herramientas sugeridas	5
Conclusiones	6
Bibliografía	6

Introducción

El diseño de la arquitectura de software es un proceso fundamental dentro del desarrollo de sistemas informáticos. Esta arquitectura define las estructuras organizacionales del software, los patrones que se utilizarán y cómo los componentes interactúan entre sí y con el entorno. En el marco del proyecto "Jacquin Academia Musical", se implementa una arquitectura orientada a objetos complementada con patrones de diseño que permiten una mayor escalabilidad, mantenibilidad y claridad del sistema.

Incorporación de patrones de diseño

Para el desarrollo del software se seleccionaron los siguientes patrones de diseño:

- **Patrón Fachada (Facade):** Centraliza las operaciones del sistema, ocultando la complejidad y ofreciendo una interfaz única de acceso a subsistemas como gestión de usuarios, cursos y aulas.
- **Patrón Estrategia (Strategy):** Permite definir diferentes algoritmos para el cálculo de matrícula o asignación de horarios, dependiendo del tipo de usuario.
- **Patrón Observador (Observer):** Implementado para mantener informados a los usuarios sobre cambios en horarios o inscripciones mediante eventos y notificaciones.

Estos patrones fortalecen la modularidad del sistema, la separación de responsabilidades y facilitan futuras modificaciones.

Vista de componentes

La vista de componentes muestra los principales bloques funcionales del sistema:

- **Componente de Autenticación:** Manejo de inicio de sesión, roles y permisos.
- **Componente Académico:** Gestión de cursos, horarios, profesores y aulas.
- **Componente Estudiantes:** Registro, inscripción, pagos y consulta de progreso.
- **Componente Administrativo:** Registro de usuarios, asignación de profesores y control general del sistema.
- **Componente de Comunicaciones:** Enlace con correo, notificaciones, mensajes internos.

Cada componente puede ser encapsulado en un módulo independiente para su despliegue y mantenimiento.

Vista de despliegue

Se propone una arquitectura cliente-servidor distribuida:

- **Cliente Web:** Interfaz construida en HTML/CSS/JavaScript que interactúa con los servicios mediante REST API.
- **Servidor Aplicativo:** Implementado en Python (Django/Flask) o Java (Spring), alojado en un servidor a elección.
- **Base de Datos:** Sistema de gestión como MySQL o PostgreSQL.
- **Servidor de Correo:** Para la notificación de eventos relevantes.
- **Opcional:** Servidor de archivos para materiales didácticos.

Esta vista permite prever las condiciones necesarias de implementación, incluyendo conectividad, disponibilidad y escalabilidad.

Herramientas sugeridas

- **Modelado:** Draw.io, StarUML, Visual Paradigm.
- **Programación:** VSCode, Eclipse, PyCharm.
- **Base de datos:** MySQL Workbench, PgAdmin.
- **Control de versiones:** Git y GitHub.
- **Documentación:** Notion, Word, Google Docs.

Conclusiones

El diseño de la arquitectura de software basado en patrones permite mejorar la calidad, escalabilidad y mantenibilidad del sistema. La selección de patrones adecuados como Fachada, Estrategia y Observador está alineada con los objetivos del proyecto de "Jacquin Academia Musical", garantizando una implementación ordenada y eficiente. Además, las vistas de componentes y despliegue facilitan la comprensión y planificación de fases futuras.

Bibliografía

- **Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1994). Design Patterns: Elements of Reusable Object-Oriented Software.** Addison-Wesley.
DOI: 10.5555/186897 [en.wikipedia.org+10dl.acm.org+10scirp.org+10](#)
- **Buschmann, F., et al. (1996). Pattern-Oriented Software Architecture Vol. 1: A System of Patterns.** Wiley.
(referencia general de la serie POSA) [en.wikipedia.org+1en.wikipedia.org+1](#)
- **“Design pattern” (Definición y contexto).** Wikipedia.
URL con referencia breve: <https://w.wiki/3XZe> [en.wikipedia.org](#)
- **SENA. (2025). Material formativo del programa Tecnólogo en Análisis y Desarrollo de Software.**
Fuente oficial del componente formativo.